2023012707 杜卓

一、操作系统引论

一、1、管理计算机硬件和软件的软件系统.

4、首先在平衡的时间限制内对外部事件做出响应的操作系统，分为硬实时和软实时.

5、为个进程共享资源时，必须保证某一时刻只有一个进程能访问该资源，以避免冲突.

二、1、硬件系统 软件系统.

3、硬件资源 软件资源

5、共享性，异步性.

二、进程的基本概述与控制

一、3、进程申请访问共享资源的代码段，需为开执行.

4、协调进程间执行顺序，以避免冲突.

三、1、动态性，异步性，结构性

5、动态 静态

6、直接制约

8、直接通信，间接通信，共享存储

9、就绪 阻塞 执行

五、2、S1是S2，S3的前驱，S2S3是S4，S5，S6的前驱，S4，S5，S6是S7的前驱.

```
3、semaphore mutex=1;
   semaphore apple=0;
   semaphore orange=0;
   semaphore empty=5;
   while(true){
      wait(empty);
      wait(mutex);
      if(放入苹果){ signal(apple);}
      else signal(orange);
      signal(mutex);
   }
```

```
while(true){
   wait(orange);
   wait(mutex);
   signal(empty);
   signal(mutex);
}
while(true){
   wait(apple);
   wait(mutex);
   signal(empty);
   signal(mutex); }
```

4、semaphore empty1=1;
semaphore full1=0;
semaphore empty2=1;
semaphore full2=0;
while(true){
wait(empty1); signal(full1);}
while(true){
wait(full1); wait(empty2); signal(empty1); signal(full2);}

while(true){
wait(full2); signal(empty2);
}

5、semaphore customers=0;
semaphore barber=0;
semaphore mutex=1;
int waiting=0;
const int N=...;
while(true){
wait(customers);
wait(mutex);
waiting--;
signal(barber);
signal(mutex);}

wait(mutex);
if(waiting<N){
waiting++;
signal(customers);
signal(mutex);
wait(barber);}
else{ signal(mutex);}

§三、处理机调度与死锁

一、2、操作系统将依某种算法从就绪队列中选择一个进程给其分配处理机的过程。

3、从作业提交到作业完成所经历的时间。

4、多个~~进程~~因相互等待资源而无法推进的状态。

进程

二、2、提交 后备 完成

6、死锁预防，死锁避免，死锁检测，死锁解除。

五、1、FCFS：

进程顺序：A(0-3)，B(3-9)，C(9-13)，D(13-18)，E(18-20)。

完成时间：A:3，B:9，C:13，D:18，E:20。

周转时间：A:3, B:7, C:9, D:13, E:12.

带权周转时间：A:1, B:1.17, C:2.25, D:2.4, E:6

平均周转时间 $\frac{3+7+9+13+12}{5}=8.6$

平均带权周转时间：(1+1.17+2.25+2.4+6)/5=2.564

非抢占SJF：

进程顺序：A(0-3), B(3-9), E到达但B已在执行不抢占, C(9-13), E(13-15), D(15-20)

完成时间：A:3, B:9, C:13, E:15, D:20.

周转时间：A:3, B:7, C:9, E:7, D:14

带权周转时间：A:1, B:1.17, C:2.25, E:3.5, D:2.8

平均周转时间：(3+7+9+7+14)/5=8

平均带权周转时间：(1+1.17+2.25+3.5+2.8)/5=2.144

抢占SJF：

进程顺序A(0-2), B到达但A剩余<B的不抢占, A(2-3), B(3-5), C到达剩余4<B的4不抢占, B(5-9), C(9-13), E到达剩余2<D的5, E(13-15), D(15-20)

完成时间：A:3, B:9, C:13, E:15, D:20

周转时间：A:3 B:7, C:9, E:7, D:14

带权周转时间：A:1, B:1.17, C:2.25, E:3.5, D:2.8

平均周转时间：8

平均带权周转时间：2.144

HRRN：

在A完成时(3)：B响应比=$\frac{1+6}{6}$=1.17. 在B完成时(9)：C响应比=$\frac{5+4}{4}$=2.25, E到达, E

在C完成时(13)：D响应比=$\frac{7+5}{5}$=2.4, E响应比=$\frac{5+2}{2}$=3.5

进程顺序：A(0-3), B(3-9), C(9-13), E(13-15), D(15-20)

完成时间：A:3, B:9, C:13, E:15, D:20

周转时间：A:3, B:7, C:9, E:7, D:14.

带权周转时间：A:1, B:1.17, C:2.25, E:3.5, D:2.8

平均周转时间：8

平均带权周转时间：2.144

PR:

进程顺序：A,B,A,B,C,A,B,C,D,B,E,C,D,B,E,D,C,D,E,D

完成时间：A：4，B：18，C：17，D：20，E：15

周转时间：A：4，B：16，C：13，D：14，E：7

带权周转时间：A：2/3，B：2/3，C：3/5，D：2/8，E：3/5

平均周转时间：(4+16+13+14+7)/5=10.8

平均带权周转时间：(1+3/3+2.6+7/8+3/5)/5=2.71

2、任务A：周期20ms，执行10ms，~~[illegible]~~

任务B：周期50ms，执行10ms，~~[illegible]~~

任务C：周期50ms，执行15ms，~~[illegible]~~

~~调度顺序：先执行~~ ~~[illegible]~~

~~根据任务到~~ ~~[illegible]~~

时间0：A：下次截止时间20，松弛=20-10-0=10

B：下次截止时间50，松弛=50-10-0=40

C：下次截止时间50，松弛=50-15-0=35 →选择A(0-10)

时间10：完A：已完成，下一个周期截止时间40.

B：松弛=50-10-10=30，C：松弛=50-15-10=25 →选C(10-25)

时间25：A：松弛=40-10-25=5，B：松弛=50-10-25=15

C：已完成，下一周期为35. →选A(25-35)

时间35：A：已完成，B：松弛=50-10-35=5 →选B(35-45)

时间45：下一个A周期截止60，B：已完成，C：35-15-45=15，→A(45-55)

A(0-10)→C(10-25)→A(25-35)→B(35-45)→A(45-55)→...

3、(1)：P0：(0,0,3,2)

P1：(1,0,0,2)

P2：(1,3,5,4)

P3：(0,3,3,2)

P4：(0,0,1,4)

work=Available=(1,6,2,2)

P0：(0,0,1,2)≤(1,6,2,2)→执行

work=(1,6,2,2)+(0,0,3,2)=(1,6,5,4)

P3：(0,6,3,2)≤(1,6,5,4)→执行

work=(1,6,5,4)+(0,3,3,2)=(1,9,8,6)

P1，P4，P2.

P0→P3→P1→P4→P2，系统安全.

(2)检查Request≤Need2→(1,2,2,2)≤(1,3,5,6)满足

检查Request≤Available→(1,2,2,2)≤(1,6,2,2)→满足

假设分配后：Available=(1,4,0,0)

Allocation2=(1,2,5,7,6)

Need2=(1,1,3,4)

执行安全性检查→可找到安全序列<P0→P3→P1→P4→P2>

可以分配.

(3)不会让死锁，当前状态为安全的

不应至少一个安全序列